

INDIAN INSTITUTE OF TECHNOLOGY INDORE

Diagonalization of General 3×3 Hermitian Matrices

M.Sc. Thesis

By

Naitik



DEPARTMENT OF PHYSICS

INDIAN INSTITUTE OF TECHNOLOGY INDORE

May 2026

Diagonalization of General 3×3 Hermitian Matrices

A THESIS

Submitted in partial fulfillment of the requirements
for the award of the degree of

Master of Science

by

Naitik



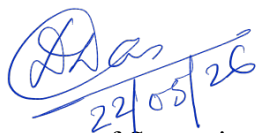
DEPARTMENT OF PHYSICS
INDIAN INSTITUTE OF TECHNOLOGY INDORE
May 2026

CANDIDATE'S DECLARATION

I hereby certify that the work presented in the thesis entitled **Diagonalization of general 3×3 Hermitian matrix** submitted in partial fulfillment of the requirements for the award of the degree of **MASTER OF SCIENCE** in the Department of PHYSICS, Indian Institute of Technology Indore, is an authentic record of my own work carried out under the supervision of Dipankar das.



Signature of Student
(Naitik)



Signature of Supervisor
(Dipankar Das)



Signature of DPGC
(Manavendra Mahato)

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my guide, **Dr. Dipankar Das**, for his continuous guidance, encouragement, support, and valuable advice throughout my academic journey. His mentorship and motivation have been a constant source of inspiration.

I am also deeply thankful to my seniors, **Anugrah Prasad** and **Shreya Pandey**, for their support, insightful discussions, and encouragement throughout this journey. Their guidance and suggestions have helped me greatly in both academic and personal growth.

I would also like to thank my M.Sc. labmate, **Shreya Soni**, for her constant support, cooperation, and helpful discussions during this period.

Finally, I express my gratitude to the Department of Physics, Indian Institute of Technology Indore, for providing a motivating academic environment and the necessary facilities.

Naitik

ABSTRACT

Diagonalization of Hermitian matrices plays an important role in quantum mechanics, neutrino oscillation studies, and computational linear algebra. In particle physics, Hermitian mass matrices arise naturally in the description of neutrino mixing phenomena, where the extraction of physical eigenvalues and mixing parameters requires stable and accurate diagonalization procedures.

In the present work, a Newton–style iterative framework for the diagonalization of Hermitian 3×3 matrices has been developed and implemented computationally. The method reformulates the diagonalization problem as a nonlinear iterative system based on unitary transformations parameterized through rotation angles and complex phase parameters. Numerical Jacobian evaluation, least-squares based correction methods, and damping techniques were incorporated to improve convergence stability and computational robustness.

The framework was implemented using *Wolfram Mathematica* and applied to Hermitian neutrino mass matrices constructed from experimentally measured neutrino oscillation parameters. The iterative procedure converged successfully toward diagonal form while preserving the Hermitian structure and unitarity of the transformation matrix. The obtained eigenvalues showed excellent agreement with standard built-in numerical diagonalization methods, confirming the accuracy of the developed approach.

The results demonstrate that the proposed iterative framework provides a stable and computationally reliable method for the diagonalization of Hermitian matrices relevant to neutrino phenomenology and computational particle physics.

Contents

Candidate's Declaration	1
Acknowledgements	2
Abstract	3
Nomenclature	1
1 Introduction	2
1.1 Background	2
1.2 Motivation	3
2 Review of Past Work and Problem Formulation	5
2.1 Hermitian Matrices in Quantum Mechanics	5
2.2 Neutrino Mixing and Mass Matrices	6
2.3 Existing Diagonalization Techniques	8
2.4 Limitations of Existing Methods	11
2.5 Problem Formulation	12
3 Methodology	14
3.1 Construction of the Hermitian Matrix	14
3.2 Mathematical Formulation of the Iterative Framework	16
3.3 Iterative Elimination of Off-Diagonal Elements	18
3.4 Newton–Style Iterative Framework	20
3.5 Algorithm Development	22
3.6 Computational Implementation	25
3.7 Flowchart of the Algorithm	25
4 Numerical Results and Analysis	27
4.1 Construction of the Neutrino Hermitian Matrix	27
4.2 Convergence Behavior of the Newton Algorithm	28
4.3 Diagonalization Results	30
4.4 Validation with Built-in Eigenvalue Methods	32
4.4.1 Verification of Unitarity	33

4.5	Physical Interpretation of Results	34
5	Conclusions and Scope for Future Work	36
5.1	Summary of the Work	36
5.2	Major Conclusions	36
A	Complete Mathematica Implementation	39
A.1	Complete Mathematica Source Code	39
A.2	Numerical Results	46

List of Figures

- 3.1 Flowchart of the Newton–style iterative diagonalization algorithm for Hermitian 3×3 matrices. 26
- 4.1 Convergence behavior of the Newton–style iterative diagonalization algorithm showing the decrease of the residual norm with iteration number. 30

List of Tables

2.1	Comparison of commonly used Hermitian matrix diagonalization methods.	10
4.1	Comparison of eigenvalues obtained using the Newton iterative framework and built-in Mathematica routines.	32

Nomenclature

H	Hermitian neutrino mass matrix
M	Majorana neutrino mass matrix
U	Unitary transformation matrix
D	Diagonal eigenvalue matrix
λ_i	Eigenvalues of the Hermitian matrix
R_{12}, R_{13}, R_{23}	Rotation matrices in two-dimensional subspaces
t_1, t_2, t_3	Rotation angle parameters
d_1, d_2, d_3	Complex phase parameters
$F(p)$	Residual vector of nonlinear equations
$J(p)$	Jacobian matrix of the residual system
Δp	Newton correction vector
α	Damping or line-search parameter
$\theta_{12}, \theta_{13}, \theta_{23}$	Neutrino mixing angles
δ_{CP}	CP-violating phase
Δm_{21}^2	Solar neutrino mass-squared difference
Δm_{31}^2	Atmospheric neutrino mass-squared difference
I	Identity matrix
$\dagger$	Hermitian conjugation operator
$\ \cdot \ $	Euclidean norm

Chapter 1

Introduction

1.1 Background

The study of matrices and their diagonalization forms an important part of modern theoretical physics, quantum mechanics, and computational linear algebra. In many physical systems, observable quantities are represented through matrices whose eigenvalues and eigenvectors correspond to measurable physical parameters. Among these, Hermitian matrices occupy a particularly important position because they possess real eigenvalues and orthogonal eigenvectors, properties that are essential for the physical interpretation of quantum mechanical systems.

In particle physics, Hermitian matrices frequently arise in the study of neutrino oscillations and mass mixing phenomena. The transition between neutrino flavor states and mass eigenstates is described through unitary transformations obtained from the diagonalization of neutrino mass matrices and effective Hamiltonians. Consequently, the determination of neutrino masses, mixing angles, and CP-violating phases depends strongly on the accurate diagonalization of complex Hermitian systems.

For three-generation neutrino models, the corresponding mixing framework is commonly represented through 3×3 matrices. The diagonalization of such matrices provides access to physically relevant quantities associated with neutrino oscillation experiments and flavor transitions. Since these systems involve complex matrix structures and very small mass scales, stable numerical diagonalization becomes an important computational problem in neutrino phenomenology.

Several analytical and numerical techniques have been developed for matrix diagonalization. Classical numerical approaches such as the Jacobi method, QR decomposition, and Householder transformations are widely used in computational linear algebra, while analytical methods based on characteristic polynomial techniques are often employed for low-dimensional systems. Although these methods are highly effective, iterative formulations remain useful for studying convergence behavior, transformation dynamics, and numerical stability associated with matrix-based physical systems.

From a computational perspective, iterative diagonalization methods provide an alternative framework in which off-diagonal matrix elements are systematically reduced through successive transformations. Such approaches are particularly useful for understanding how convergence properties depend on matrix structure, parameterization schemes, and numerical correction procedures. In low-dimensional Hermitian systems relevant to particle physics, these methods also provide a direct connection between mathematical transformations and physically observable quantities.

Motivated by these considerations, the present work focuses on the development and analysis of an iterative framework for the diagonalization of Hermitian 3×3 matrices with applications to neutrino mass systems and computational particle physics.

1.2 Motivation

Although several analytical and numerical techniques exist for the diagonalization of Hermitian matrices, the study of iterative nonlinear approaches remains important for understanding convergence behavior, numerical stability, and transformation dynamics in matrix-based physical systems. In many computational problems arising in particle physics, the accuracy and stability of the diagonalization procedure play a crucial role in determining physically meaningful quantities.

In neutrino phenomenology, the extraction of neutrino mass eigenvalues and mixing parameters requires the diagonalization of complex Hermitian matrices constructed from experimentally measured oscillation data. Since these systems involve very small mass scales and complex phase structures, numerical instabilities may arise during matrix transformations and iterative computations. Consequently, the development of computationally stable diagonalization frameworks remains an important problem in theoretical and computational physics.

Most standard numerical methods primarily focus on obtaining the final eigenvalue spectrum efficiently. However, iterative formulations based on successive transformations provide additional insight into the evolution of matrix structure during the diagonalization process. Such approaches are useful for analyzing how off-diagonal elements are progressively suppressed and how convergence properties depend on parameterization schemes and correction procedures.

The motivation for the present work therefore arises from the need to formulate a systematic iterative framework capable of diagonalizing Hermitian 3×3 matrices while preserving numerical stability and unitarity. By combining unitary parameterization with Newton-style iterative correction methods, the work aims to study both the mathematical structure and computational behavior of the diagonalization process.

Another important motivation of this study is to investigate the applicability of iterative nonlinear techniques to matrix problems relevant to particle physics. The develop-

ment of stable computational procedures for low-dimensional Hermitian systems may contribute toward future studies involving neutrino mass models, quantum systems, and numerical approaches in computational particle physics.

Chapter 2

Review of Past Work and Problem Formulation

2.1 Hermitian Matrices in Quantum Mechanics

Hermitian matrices play a central role in quantum mechanics and modern theoretical physics because they provide the mathematical representation of physical observables. In quantum systems, measurable quantities such as energy, momentum, angular momentum, and effective Hamiltonians are represented through Hermitian operators whose matrix representations satisfy specific symmetry properties. The Hermitian structure guarantees real eigenvalue spectra and unitary diagonalization, properties that are essential for the physical interpretation of observable systems.

A matrix H is said to be Hermitian if it satisfies the condition

$$H^\dagger = H, \quad (2.1)$$

where $H^\dagger$ denotes the conjugate transpose of the matrix H . For a general complex matrix,

$$H = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix}, \quad (2.2)$$

the Hermiticity condition requires

$$h_{ij} = h_{ji}^*, \quad (2.3)$$

where $*$ represents complex conjugation. Consequently, the diagonal elements of a Hermitian matrix are always real, while the off-diagonal elements occur as complex conjugate pairs.

One of the most important properties of Hermitian matrices is the existence of real

eigenvalues. If

$$H\psi = \lambda\psi, \quad (2.4)$$

then the eigenvalue λ must be real. This property is fundamentally important in quantum mechanics because physical observables correspond to experimentally measurable quantities and therefore cannot possess complex eigenvalues.

Hermitian matrices also possess orthogonal eigenvectors that form a complete basis in the corresponding vector space. As a result, any Hermitian matrix can be diagonalized through a unitary transformation of the form

$$D = U^\dagger H U, \quad (2.5)$$

where U is a unitary matrix satisfying

$$U^\dagger U = I, \quad (2.6)$$

and D is a diagonal matrix containing the eigenvalues of H . The preservation of unitarity during transformation is particularly important in quantum mechanical systems because it ensures conservation of probability and orthogonality of physical states.

In particle physics, Hermitian matrices arise naturally in neutrino oscillation studies, flavor mixing problems, and effective Hamiltonian formulations. The diagonalization of these matrices allows the transition from interaction or flavor bases to physical mass eigenstate bases, making matrix diagonalization an essential computational step in extracting physically observable quantities.

For low-dimensional systems such as 3×3 neutrino mass matrices, diagonalization procedures provide direct access to neutrino mass eigenvalues, mixing parameters, and CP-violating phases. Since such systems often involve complex phase structures and small mass scales, the development of stable and computationally reliable diagonalization methods remains an important problem in computational particle physics.

2.2 Neutrino Mixing and Mass Matrices

The discovery of neutrino oscillations established that neutrinos possess non-zero masses and that neutrino flavor eigenstates differ from mass eigenstates. This phenomenon cannot be explained within the original formulation of the Standard Model and therefore provides important evidence for physics beyond the Standard Model. The mathematical description of neutrino oscillation phenomena naturally leads to the study of complex mass matrices and their diagonalization.

In neutrino physics, the flavor eigenstates are represented as

$$\nu_f = \begin{pmatrix} \nu_e \\ \nu_\mu \\ \nu_\tau \end{pmatrix}, \quad (2.7)$$

while the corresponding mass eigenstates are written as

$$\nu_m = \begin{pmatrix} \nu_1 \\ \nu_2 \\ \nu_3 \end{pmatrix}. \quad (2.8)$$

The relation between the flavor basis and mass basis is described through the Pontecorvo–Maki–Nakagawa–Sakata (PMNS) mixing matrix,

$$\nu_f = U_{PMNS} \nu_m. \quad (2.9)$$

Explicitly, the transformation takes the form

$$\begin{pmatrix} \nu_e \\ \nu_\mu \\ \nu_\tau \end{pmatrix} = \begin{pmatrix} U_{e1} & U_{e2} & U_{e3} \\ U_{\mu1} & U_{\mu2} & U_{\mu3} \\ U_{\tau1} & U_{\tau2} & U_{\tau3} \end{pmatrix} \begin{pmatrix} \nu_1 \\ \nu_2 \\ \nu_3 \end{pmatrix}. \quad (2.10)$$

The PMNS matrix is unitary and satisfies

$$U_{PMNS}^\dagger U_{PMNS} = I. \quad (2.11)$$

Its matrix elements contain the neutrino mixing angles and CP-violating phase parameters that govern oscillation probabilities.

The neutrino mass matrix contains the information associated with neutrino masses and flavor mixing. In general, the mass matrix may possess complex off-diagonal terms representing coupling between different flavor states. For a complex Hermitian neutrino mass matrix,

$$M = \begin{pmatrix} a_{11} & a_{12} + ib_{12} & a_{13} + ib_{13} \\ a_{12} - ib_{12} & a_{22} & a_{23} + ib_{23} \\ a_{13} - ib_{13} & a_{23} - ib_{23} & a_{33} \end{pmatrix} \quad (2.12)$$

where the diagonal elements a_{11}, a_{22}, a_{33} are real, and the off-diagonal elements satisfy the Hermitian condition

$$M_{ij} = M_{ji}^*.$$

the physical neutrino masses are obtained through unitary diagonalization,

$$D = U^\dagger M U, \quad (2.13)$$

where

$$D = \begin{pmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{pmatrix} \quad (2.14)$$

contains the neutrino mass eigenvalues.

The eigenvalues of the matrix are determined from the characteristic equation

$$\det(M - \lambda I) = 0. \quad (2.15)$$

For a 3×3 system, this equation reduces to a cubic polynomial in λ , whose solutions correspond to the physical neutrino masses. The corresponding eigenvectors determine the mixing structure encoded in the PMNS matrix.

In computational neutrino physics, the diagonalization of such matrices is essential for studying oscillation probabilities, mass hierarchies, and CP-violation effects. Since neutrino mass matrices generally involve complex phases and small mass scales, stable numerical diagonalization methods become important for both analytical investigations and computational simulations.

Consequently, the study of iterative approaches for Hermitian matrix diagonalization remains relevant for understanding the numerical behavior, convergence properties, and computational stability associated with neutrino mass systems and flavor mixing problems.

2.3 Existing Diagonalization Techniques

The diagonalization of Hermitian matrices has been extensively studied in numerical linear algebra, quantum mechanics, and computational physics because of its direct connection with eigenvalue problems and physical observables. Over the years, several analytical and numerical methods have been developed for obtaining the eigenvalues and eigenvectors of Hermitian systems. These approaches differ in terms of computational efficiency, numerical stability, convergence behavior, and applicability to complex physical systems.

One of the earliest iterative approaches for Hermitian matrix diagonalization is the Jacobi method. In this method, successive unitary rotations are applied to eliminate off-diagonal elements of the matrix. For a Hermitian matrix H , the transformation is written as

$$H' = R^\dagger H R, \quad (2.16)$$

where R represents a rotation matrix chosen such that a selected off-diagonal element becomes zero after transformation. Repeated application of these rotations gradually reduces the matrix toward diagonal form. The Jacobi method is known for its numerical stability and high accuracy, particularly for low-dimensional Hermitian systems, although the number of iterations required may become computationally expensive for repeated calculations.

Another widely used approach is the QR algorithm, which is based on orthogonal matrix factorization. In this method, the matrix is decomposed as

$$H = QR, \quad (2.17)$$

where Q is an orthogonal or unitary matrix and R is an upper triangular matrix. A new matrix is then constructed through

$$H_1 = RQ. \quad (2.18)$$

The process is repeated iteratively according to

$$H_{k+1} = R_k Q_k, \quad (2.19)$$

until the matrix converges toward diagonal form. QR-based techniques are computationally efficient and widely used in standard numerical eigensolver libraries because of their strong convergence properties.

Householder transformation methods are also frequently employed in Hermitian eigenvalue problems. These transformations reduce the original Hermitian matrix into tridiagonal form through unitary similarity transformations,

$$H' = P^\dagger H P, \quad (2.20)$$

where P is a Householder matrix. The resulting tridiagonal matrix is computationally simpler to diagonalize and is often combined with QR-type algorithms for efficient eigenvalue computation.

For low-dimensional systems, analytical methods based on characteristic polynomial techniques are also commonly used. The eigenvalues are obtained from the characteristic equation

$$\det(H - \lambda I) = 0. \quad (2.21)$$

For a 3×3 Hermitian matrix, this equation reduces to a cubic polynomial in λ ,

$$\lambda^3 + p\lambda^2 + q\lambda + r = 0. \quad (2.22)$$

Analytical approaches based on Cardanos formula can then be used to determine the eigenvalues explicitly. Although such methods provide exact mathematical solutions, the resulting expressions may become algebraically complicated and numerically unstable when the matrix elements differ significantly in magnitude.

In addition to these classical approaches, iterative nonlinear methods have also been explored for solving matrix eigenvalue problems. Such methods reformulate diagonalization as a residual minimization or root-finding problem and iteratively update transformation parameters until convergence is achieved. These approaches are useful for studying transformation dynamics, convergence behavior, and stability properties associated with matrix diagonalization procedures.

A qualitative comparison between some commonly used diagonalization methods is summarized in Table ??.

Table 2.1: Comparison of commonly used Hermitian matrix diagonalization methods.

Method	Main Idea	Advantages	Limitations
Jacobi Method [1]	Successive rotational elimination	Stable and accurate	Large number of iterations
QR Algorithm [2]	Orthogonal matrix factorization	Fast convergence	Less interpretive structure
Householder Method [1]	Reduction to tridiagonal form	Efficient preprocessing	Requires additional transformations
Analytical Methods [3]	Characteristic polynomial solutions	Exact eigenvalues	Algebraically complicated
Iterative Nonlinear Methods [4]	Residual minimization	Flexible and systematic	Convergence sensitive

Although these methods are highly effective for eigenvalue computation, studying iterative frameworks remains important for understanding how matrix structure evolves during transformation and how convergence behavior depends on parameterization and correction schemes. Such considerations are particularly relevant in computational particle physics, where Hermitian matrices involving complex phase structures and small mass scales frequently arise.

2.4 Limitations of Existing Methods

Although the diagonalization techniques discussed in the previous section are widely used in computational physics and numerical linear algebra, each method possesses certain mathematical and computational limitations when applied to complex Hermitian systems. These limitations become particularly relevant in particle physics applications, where matrix structures often involve complex phases, small mass scales, and repeated numerical computations.

Analytical approaches based on characteristic polynomial techniques provide exact mathematical expressions for eigenvalues. However, for 3×3 Hermitian systems, the resulting cubic equations may become algebraically complicated and computationally sensitive to numerical precision. Small variations in matrix elements can lead to instability during floating-point calculations, especially when the eigenvalues differ significantly in magnitude.

Iterative rotational methods such as the Jacobi technique are known for their numerical stability and accuracy. Nevertheless, the method requires repeated elimination of off-diagonal elements through successive rotations, which may increase computational cost during repeated calculations or parameter scans. In addition, the convergence rate may become slow when the matrix contains strongly coupled off-diagonal terms.

QR-based methods are computationally efficient and widely implemented in standard eigensolver libraries. However, these approaches primarily focus on rapid eigenvalue computation rather than analysis of the intermediate transformation behavior during diagonalization. Consequently, the evolution of matrix structure and suppression of off-diagonal terms are not always directly accessible within the computational procedure.

Householder reduction methods efficiently transform Hermitian matrices into tridiagonal form, but they generally require additional transformation steps before complete diagonalization is achieved. For low-dimensional systems, such preprocessing procedures may introduce unnecessary computational overhead compared to more direct iterative formulations.

Another important limitation of many standard numerical approaches is that they are designed as general-purpose eigensolvers rather than frameworks specifically tailored for studying iterative transformation dynamics. In problems involving neutrino mass matrices and flavor mixing systems, it is often useful to analyze how convergence behavior depends on parameterization schemes, correction procedures, and matrix structure during the diagonalization process.

Furthermore, Hermitian matrices arising in particle physics frequently contain complex phase structures and nearly singular numerical configurations associated with very small neutrino mass scales. Such systems may produce ill-conditioned numerical be-

havior during iterative computations, making stability and convergence control important aspects of the diagonalization procedure.

These limitations motivate the development of alternative iterative frameworks capable of systematically reducing off-diagonal matrix elements while preserving numerical stability and unitary structure throughout the transformation process. The present work therefore focuses on the formulation of a Newton–style iterative approach for the diagonalization of Hermitian 3×3 matrices relevant to neutrino phenomenology and computational particle physics.

2.5 Problem Formulation

The diagonalization of a Hermitian matrix may be formulated as the problem of determining a unitary transformation that eliminates all off-diagonal elements of the matrix while preserving its eigenvalue spectrum. For a Hermitian 3×3 matrix H ,

$$H = \begin{pmatrix} H_{11} & H_{12} & H_{13} \\ H_{12}^* & H_{22} & H_{23} \\ H_{13}^* & H_{23}^* & H_{33} \end{pmatrix}, \quad (2.23)$$

the objective is to determine a unitary matrix U such that

$$D = U^\dagger H U, \quad (2.24)$$

where D is a diagonal matrix containing the eigenvalues of H ,

$$D = \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{pmatrix}. \quad (2.25)$$

Instead of directly solving the characteristic equation, the present work reformulates the diagonalization problem as a nonlinear iterative system. The unitary transformation matrix is parameterized through successive complex rotations involving three rotation angles and three phase parameters. The transformation matrix may therefore be written as

$$U = R_{23}(t_3, d_3) R_{13}(t_2, d_2) R_{12}(t_1, d_1), \quad (2.26)$$

where t_i represent rotation angles and d_i denote complex phase parameters associated with the corresponding unitary rotations.

After applying the transformation, the transformed matrix becomes

$$M = U^\dagger H U. \quad (2.27)$$

For complete diagonalization, all off-diagonal elements of the transformed matrix must vanish. Consequently, the diagonalization conditions are expressed as

$$M_{12} = 0, \quad M_{13} = 0, \quad M_{23} = 0. \quad (2.28)$$

Since the matrix elements are generally complex, both the real and imaginary parts of the off-diagonal terms must independently satisfy the diagonalization conditions. The problem therefore reduces to solving the nonlinear residual system

$$F(p) = \begin{pmatrix} \text{Re}(M_{12}) \\ \text{Im}(M_{12}) \\ \text{Re}(M_{13}) \\ \text{Im}(M_{13}) \\ \text{Re}(M_{23}) \\ \text{Im}(M_{23}) \end{pmatrix} = 0, \quad (2.29)$$

where

$$p = (t_1, t_2, t_3, d_1, d_2, d_3) \quad (2.30)$$

represents the vector of unknown transformation parameters.

The diagonalization problem is therefore converted into a coupled nonlinear root-finding problem in six variables. Solving this system iteratively allows the off-diagonal elements of the Hermitian matrix to be progressively suppressed until convergence toward diagonal form is achieved.

Motivated by this formulation, the following chapter develops a Newton-style iterative framework for determining the transformation parameters and studying the convergence behavior associated with Hermitian matrix diagonalization.

Chapter 3

Methodology

3.1 Construction of the Hermitian Matrix

In particle physics, neutrino oscillation phenomena are described through neutrino mass and mixing parameters obtained from experimental observations. The mathematical structure associated with these systems naturally leads to the appearance of complex Hermitian matrices whose diagonalization provides physically observable quantities such as neutrino mass eigenvalues and mixing parameters. In the present work, the construction of the Hermitian matrix forms the starting point of the iterative diagonalization framework. [3]

The analysis begins with experimentally measured neutrino oscillation parameters. These include the three neutrino mixing angles,

$$\theta_{12}, \quad \theta_{23}, \quad \theta_{13}, \quad (3.1)$$

the CP-violating phase,

$$\delta_{CP}, \quad (3.2)$$

and the experimentally measured mass-squared differences,

$$\Delta m_{21}^2, \quad \Delta m_{31}^2. \quad (3.3)$$

Using the normal mass ordering approximation, the neutrino masses are reconstructed as

$$m_1 \approx 0, \quad (3.4)$$

$$m_2 = \sqrt{\Delta m_{21}^2}, \quad (3.5)$$

$$m_3 = \sqrt{\Delta m_{31}^2}. \quad (3.6)$$

These masses are then used to construct the diagonal mass matrix,

$$D = \begin{pmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{pmatrix}. \quad (3.7)$$

The flavor mixing structure of neutrinos is described by the PMNS mixing matrix U_{PMNS} , which relates the flavor eigenstates to the mass eigenstates. Using the mixing angles and CP phase, the PMNS matrix is parameterized as

$$U_{PMNS} = \begin{pmatrix} c_{12}c_{13} & s_{12}c_{13} & s_{13}e^{-i\delta_{CP}} \\ -s_{12}c_{23} - c_{12}s_{23}s_{13}e^{i\delta_{CP}} & c_{12}c_{23} - s_{12}s_{23}s_{13}e^{i\delta_{CP}} & s_{23}c_{13} \\ s_{12}s_{23} - c_{12}c_{23}s_{13}e^{i\delta_{CP}} & -c_{12}s_{23} - s_{12}c_{23}s_{13}e^{i\delta_{CP}} & c_{23}c_{13} \end{pmatrix}, \quad (3.8)$$

where

$$c_{ij} = \cos \theta_{ij}, \quad s_{ij} = \sin \theta_{ij}. \quad (3.9)$$

Using the PMNS matrix and the diagonal mass matrix, the Majorana neutrino mass matrix is constructed as

$$M = U_{PMNS}^* D U_{PMNS}^T. \quad (3.10)$$

To ensure a Hermitian structure suitable for numerical diagonalization, the matrix

$$H = M^\dagger M \quad (3.11)$$

is constructed. By definition,

$$H^\dagger = H, \quad (3.12)$$

which guarantees that H is Hermitian. Consequently, all eigenvalues of H are real and non-negative.

Using the NuFIT 6.0 experimental data employed in the computational implementation, the resulting Hermitian matrix obtained in the present work is of the form [5]

$$H = \begin{pmatrix} 7.81988 & -18.8279 - 13.3193i & -24.9634 - 14.1341i \\ -18.8279 + 13.3193i & 118.786 & 120.113 + 0.272628i \\ -24.9634 + 14.1341i & 120.113 - 0.272628i & 132.184 \end{pmatrix}. \quad (3.13)$$

The eigenvalues of this Hermitian matrix correspond to the neutrino mass-squared values and are obtained through the diagonalization procedure developed in the present work. The entire iterative framework described in the following sections is therefore constructed around transforming this Hermitian matrix into diagonal form through successive unitary transformations.

3.2 Mathematical Formulation of the Iterative Framework

The primary objective of the present work is to diagonalize the Hermitian matrix H through a suitable unitary transformation. For a Hermitian matrix, diagonalization is achieved by finding a unitary matrix U such that

$$U^\dagger H U = D, \quad (3.14)$$

where D is a diagonal matrix containing the eigenvalues of H ,

$$D = \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{pmatrix}. \quad (3.15)$$

The matrix U must satisfy the unitarity condition

$$U^\dagger U = I. \quad (3.16)$$

Instead of directly solving the characteristic equation analytically, the present work formulates the diagonalization problem as a nonlinear system involving rotation and phase parameters. The unitary transformation matrix is parameterized as a product of successive rotations in different two-dimensional subspaces.[6]

The unitary matrix is written in the form

$$U = R_{23}(t_3, d_3) R_{13}(t_2, d_2) R_{12}(t_1, d_1), \quad (3.17)$$

where t_1, t_2, t_3 represent rotation angles and d_1, d_2, d_3 denote complex phase parameters.

The rotation in the $(1, 2)$ plane is represented as

$$R_{12}(t_1, d_1) = \begin{pmatrix} \cos t_1 & \sin t_1 e^{id_1} & 0 \\ -\sin t_1 e^{-id_1} & \cos t_1 & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (3.18)$$

Similarly, the rotation matrices in the $(1, 3)$ and $(2, 3)$ planes are given by

$$R_{13}(t_2, d_2) = \begin{pmatrix} \cos t_2 & 0 & \sin t_2 e^{id_2} \\ 0 & 1 & 0 \\ -\sin t_2 e^{-id_2} & 0 & \cos t_2 \end{pmatrix}, \quad (3.19)$$

and

$$R_{23}(t_3, d_3) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos t_3 & \sin t_3 e^{id_3} \\ 0 & -\sin t_3 e^{-id_3} & \cos t_3 \end{pmatrix}. \quad (3.20)$$

The transformed matrix is then constructed as

$$M = U^\dagger H U. \quad (3.21)$$

For exact diagonalization, all off-diagonal elements of M must vanish. Therefore, the following conditions are imposed:

$$M_{12} = 0, \quad M_{13} = 0, \quad M_{23} = 0. \quad (3.22)$$

Since the matrix elements are complex quantities, each condition produces two independent equations corresponding to the real and imaginary parts,

$$\text{Re}(M_{ij}) = 0, \quad \text{Im}(M_{ij}) = 0. \quad (3.23)$$

Consequently, the diagonalization problem reduces to solving a coupled nonlinear system consisting of six equations in six unknown parameters. The parameter vector is defined as

$$p = (t_1, t_2, t_3, d_1, d_2, d_3). \quad (3.24)$$

The residual vector associated with the nonlinear system is written as

$$F(p) = \begin{pmatrix} \text{Re}(M_{12}) \\ \text{Im}(M_{12}) \\ \text{Re}(M_{13}) \\ \text{Im}(M_{13}) \\ \text{Re}(M_{23}) \\ \text{Im}(M_{23}) \end{pmatrix}. \quad (3.25)$$

The objective of the iterative procedure is therefore to determine the parameter vector p such that

$$F(p) = 0. \quad (3.26)$$

When this condition is satisfied, all off-diagonal elements vanish and the transformed matrix becomes diagonal. Thus, the matrix diagonalization problem is reformulated as a nonlinear root-finding problem whose solution determines the required unitary transformation matrix.

3.3 Iterative Elimination of Off-Diagonal Elements

The central idea of the present work is to achieve diagonalization of the Hermitian matrix through the systematic elimination of off-diagonal elements using an iterative procedure. Instead of directly solving for the eigenvalues analytically, the method progressively adjusts the transformation parameters so that the off-diagonal components of the transformed matrix approach zero.

Starting from the Hermitian matrix H , the transformed matrix is constructed through the unitary transformation

$$M = U^\dagger H U, \quad (3.27)$$

where the unitary matrix U depends on the set of rotation and phase parameters,

$$p = (t_1, t_2, t_3, d_1, d_2, d_3). \quad (3.28)$$

For exact diagonalization, the transformed matrix must satisfy

$$M_{ij} = 0 \quad \text{for } i \neq j. \quad (3.29)$$

However, for an arbitrary initial choice of parameters, the transformed matrix generally contains non-zero off-diagonal terms. Therefore, the diagonalization problem is reformulated as the problem of minimizing these off-diagonal components iteratively.

The residual vector is defined using the real and imaginary parts of the off-diagonal elements,

$$F(p) = \begin{pmatrix} \text{Re}(M_{12}) \\ \text{Im}(M_{12}) \\ \text{Re}(M_{13}) \\ \text{Im}(M_{13}) \\ \text{Re}(M_{23}) \\ \text{Im}(M_{23}) \end{pmatrix}. \quad (3.30)$$

The magnitude of the residual vector measures how far the transformed matrix is from diagonal form. The total residual error is quantified through the norm

$$\|F(p)\| = \sqrt{\sum_{i=1}^6 F_i^2}. \quad (3.31)$$

The iterative process begins with an initial guess for the parameter vector,

$$p^{(0)} = (t_1^{(0)}, t_2^{(0)}, t_3^{(0)}, d_1^{(0)}, d_2^{(0)}, d_3^{(0)}). \quad (3.32)$$

Using these parameters, the transformed matrix $M^{(0)}$ and the corresponding residual vector $F(p^{(0)})$ are computed. If the residual norm is not sufficiently small, the parameters are corrected iteratively in order to reduce the off-diagonal contributions.

At the k^{th} iteration, the parameter vector is updated as

$$p^{(k+1)} = p^{(k)} - \Delta p^{(k)}, \quad (3.33)$$

where $\Delta p^{(k)}$ represents the correction vector obtained from the Newton iteration procedure.

After every update, a new transformed matrix is constructed,

$$M^{(k+1)} = U^\dagger(p^{(k+1)}) H U(p^{(k+1)}), \quad (3.34)$$

and the corresponding residual vector is recalculated. The iteration proceeds until the convergence condition

$$\|F(p^{(k)})\| < \epsilon \quad (3.35)$$

is satisfied, where ϵ is a predefined numerical tolerance.

The iterative elimination procedure therefore gradually suppresses all off-diagonal elements of the transformed matrix,

$$M_{12}, \quad M_{13}, \quad M_{23} \rightarrow 0, \quad (3.36)$$

leading the matrix toward diagonal form. When convergence is achieved, the transformed matrix becomes approximately

$$M \approx \begin{pmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{pmatrix}, \quad (3.37)$$

where the diagonal entries correspond to the eigenvalues of the original Hermitian matrix.

This iterative reduction of off-diagonal elements forms the mathematical foundation of the diagonalization framework developed in the present work and provides a systematic approach for obtaining stable numerical solutions for Hermitian 3×3 matrices.

3.4 Newton–Style Iterative Framework

The nonlinear system obtained from the diagonalization conditions cannot generally be solved analytically because the transformed matrix elements depend nonlinearly on the rotation angles and phase parameters. Therefore, an iterative numerical procedure is required to determine the parameter vector that minimizes the off-diagonal elements of the transformed matrix.

In the present work, the coupled nonlinear system is solved using a Newton–style iterative method. The basic idea of the Newton method is to approximate the nonlinear system locally by a linear system and then compute corrections that move the solution closer to the desired root.

The nonlinear diagonalization conditions are collectively represented through the residual vector

$$F(p) = 0, \quad (3.38)$$

where

$$p = (t_1, t_2, t_3, d_1, d_2, d_3) \quad (3.39)$$

is the vector containing the unknown rotation and phase parameters.

At a given iteration point $p^{(k)}$, the residual vector is expanded using a first-order Taylor approximation,

$$F(p^{(k)} + \Delta p) \approx F(p^{(k)}) + J(p^{(k)})\Delta p, \quad (3.40)$$

where $J(p^{(k)})$ is the Jacobian matrix of the nonlinear system evaluated at the current iteration point.

The Jacobian matrix is defined as

$$J_{ij} = \frac{\partial F_i}{\partial p_j}. \quad (3.41)$$

Since the residual vector contains six equations and six unknown parameters, the Jacobian is a 6×6 matrix,

$$J(p) = \begin{pmatrix} \frac{\partial F_1}{\partial p_1} & \frac{\partial F_1}{\partial p_2} & \cdots & \frac{\partial F_1}{\partial p_6} \\ \frac{\partial F_2}{\partial p_1} & \frac{\partial F_2}{\partial p_2} & \cdots & \frac{\partial F_2}{\partial p_6} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial F_6}{\partial p_1} & \frac{\partial F_6}{\partial p_2} & \cdots & \frac{\partial F_6}{\partial p_6} \end{pmatrix}. \quad (3.42)$$

For the correct solution, the updated residual must vanish,

$$F(p^{(k)} + \Delta p) = 0. \quad (3.43)$$

Substituting the first-order approximation gives the linear system

$$J(p^{(k)})\Delta p = F(p^{(k)}). \quad (3.44)$$

Solving this equation provides the correction vector Δp . The parameters are then updated according to

$$p^{(k+1)} = p^{(k)} - \Delta p. \quad (3.45)$$

In the computational implementation, the Jacobian matrix is evaluated numerically using the finite-difference approximation, [4]

$$\frac{\partial F_i}{\partial p_j} \approx \frac{F_i(p_j + h) - F_i(p_j - h)}{2h}, \quad (3.46)$$

where h is a small numerical step size.

A direct Newton update may sometimes become unstable if the correction vector is too large. To improve convergence stability, a damping parameter or step-size factor α is introduced in the update equation,

$$p^{(k+1)} = p^{(k)} - \alpha \Delta p, \quad (3.47)$$

where

$$0 < \alpha \leq 1. \quad (3.48)$$

Initially, the full Newton step $\alpha = 1$ is attempted. If the residual norm increases after the update, the step size is reduced iteratively according to

$$\alpha \rightarrow \frac{\alpha}{2}, \quad (3.49)$$

until a reduction in the residual norm is achieved. This line-search procedure improves numerical stability and prevents divergence during the iteration process. [7]

The convergence criterion used in the present work is

$$\|F(p^{(k)})\| < 10^{-10}. \quad (3.50)$$

Once this condition is satisfied, the off-diagonal elements of the transformed matrix become numerically negligible and the matrix is effectively diagonalized.

The Newton–style iterative framework therefore provides a systematic and computationally stable procedure for solving the nonlinear diagonalization problem associated with Hermitian 3×3 matrices.

3.5 Algorithm Development

Based on the mathematical formulation and Newton–style iterative framework discussed in the previous sections, a computational algorithm was developed for the diagonalization of Hermitian 3×3 matrices. The algorithm combines unitary parameterization, residual minimization, numerical Jacobian evaluation, and iterative parameter correction to systematically eliminate the off-diagonal elements of the transformed matrix.

The complete procedure begins with the construction of the Hermitian matrix H from neutrino oscillation parameters. Using the experimentally measured mixing angles, CP phase, and mass-squared differences, the PMNS matrix and mass matrix are constructed, followed by the Hermitian matrix

$$H = M^\dagger M. \quad (3.51)$$

The unitary transformation matrix is parameterized using three rotation angles and three phase parameters,

$$p = (t_1, t_2, t_3, d_1, d_2, d_3). \quad (3.52)$$

Using these parameters, the unitary matrix is constructed as

$$U = R_{23}(t_3, d_3)R_{13}(t_2, d_2)R_{12}(t_1, d_1). \quad (3.53)$$

The transformed matrix is then evaluated through

$$M = U^\dagger H U. \quad (3.54)$$

For exact diagonalization, all off-diagonal elements of M must vanish. Therefore, the residual vector is defined using the real and imaginary parts of the off-diagonal elements,

$$F(p) = \begin{pmatrix} \text{Re}(M_{12}) \\ \text{Im}(M_{12}) \\ \text{Re}(M_{13}) \\ \text{Im}(M_{13}) \\ \text{Re}(M_{23}) \\ \text{Im}(M_{23}) \end{pmatrix}. \quad (3.55)$$

The algorithm then proceeds iteratively using the Newton update framework. At each iteration:

1. The residual vector $F(p)$ is computed.
2. The Jacobian matrix $J(p)$ is evaluated numerically using finite-difference approximation.
3. The linear system

$$J(p)\Delta p = F(p) \quad (3.56)$$

is solved to determine the correction vector Δp .

4. The parameters are updated according to

$$p_{\text{new}} = p - \alpha \Delta p, \quad (3.57)$$

where α is the damping parameter introduced for stability.

5. The transformed matrix and residual norm are recalculated.
6. The iterative procedure continues until the convergence condition

$$\|F(p)\| < \epsilon \quad (3.58)$$

is satisfied.

In the computational implementation, the Jacobian matrix occasionally becomes nearly singular due to the presence of very small neutrino masses, particularly when

$m_1 \approx 0$. To improve numerical stability, the standard linear solver was replaced with a least-squares based pseudoinverse method using singular value decomposition (SVD).

Instead of solving

$$\Delta p = J^{-1}F, \quad (3.59)$$

the correction vector is computed using the least-squares solution,

$$\Delta p = \text{LeastSquares}(J, F). \quad (3.60)$$

This modification significantly improves stability in situations where the Jacobian matrix is ill-conditioned.

To avoid convergence toward incorrect local solutions, a multi-start strategy was also incorporated into the algorithm. Multiple initial parameter guesses are tested independently, and the solution with the smallest residual norm is selected as the final converged solution.

The complete computational workflow of the developed algorithm can therefore be summarized as follows:

1. Construct the Hermitian matrix H .
2. Initialize the parameter vector p .
3. Build the unitary matrix U .
4. Compute the transformed matrix $M = U^\dagger H U$.
5. Evaluate the residual vector $F(p)$.
6. Compute the numerical Jacobian matrix.
7. Solve for the correction vector using least-squares.
8. Apply damping and update parameters.
9. Repeat until convergence.
10. Obtain the diagonal matrix and eigenvalues.

The algorithm developed in the present work therefore provides a stable and systematic framework for the numerical diagonalization of Hermitian 3×3 matrices relevant to neutrino oscillation studies and particle physics applications.

3.6 Computational Implementation

The proposed iterative diagonalization framework was implemented using *Wolfram Mathematica*. The implementation involved construction of the Hermitian neutrino mass matrix, parameterization of the unitary transformation matrix, evaluation of the residual equations, and iterative computation of the transformation parameters.

The diagonalization problem was formulated as a nonlinear root-finding system in terms of three rotation angles and three phase parameters. At each iteration step, the transformed matrix

$$M = U^\dagger H U \quad (3.61)$$

was computed, and the real and imaginary parts of its off-diagonal elements were used to construct the residual vector.

The Jacobian matrix associated with the residual system was evaluated numerically using finite-difference approximation. The correction vector was then obtained through a least-squares based Newton update procedure. To improve numerical stability during iteration, a damping and line-search method was incorporated into the parameter update scheme.

The iterative process continued until the residual norm satisfied the convergence condition

$$\|F(p)\| < 10^{-10}. \quad (3.62)$$

The complete computational workflow of the developed algorithm is summarized through the pseudocode and flowchart presented in the following sections, while the full Mathematica implementation is provided in Appendix A.

The numerical implementation demonstrated stable convergence behavior and successfully reproduced the expected neutrino mass-squared eigenvalues for the Hermitian systems considered in the present work.

3.7 Flowchart of the Algorithm

The computational workflow of the developed Newton–style iterative diagonalization framework is illustrated in Figure 3.1. The algorithm begins with the construction of the Hermitian neutrino mass matrix and proceeds iteratively through residual minimization and Newton updates until convergence is achieved.

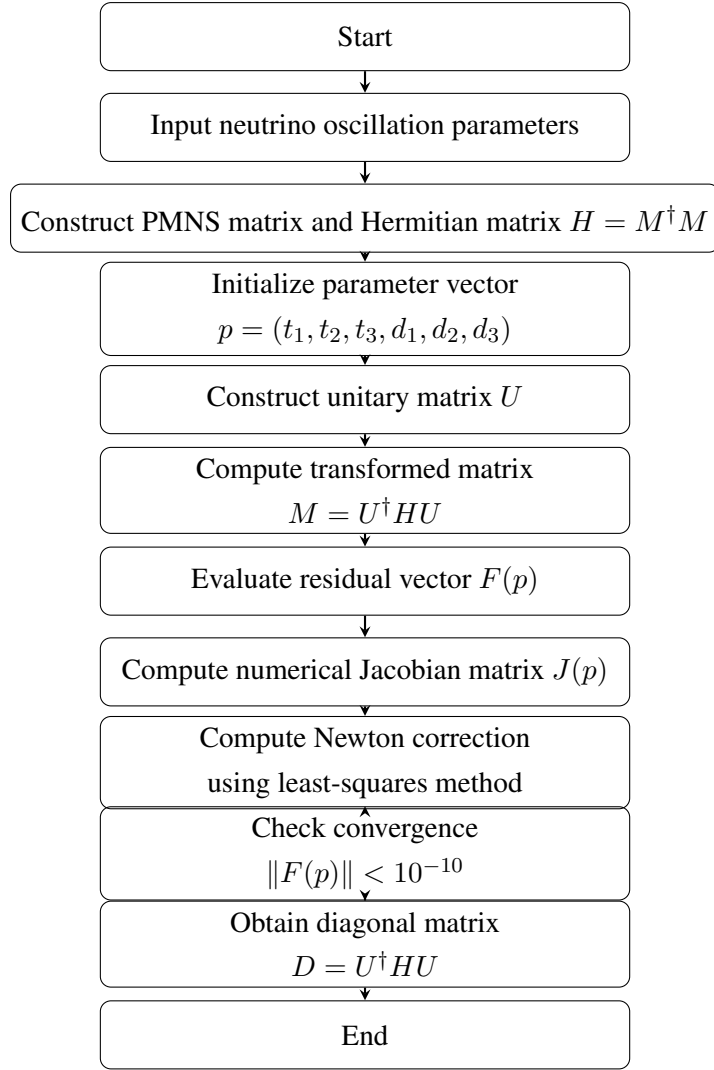


Figure 3.1: Flowchart of the Newton–style iterative diagonalization algorithm for Hermitian 3×3 matrices.

The above flowchart summarizes the computational sequence followed in the present work. Starting from experimentally measured neutrino oscillation parameters, the algorithm systematically constructs the Hermitian matrix and iteratively minimizes the off-diagonal elements until convergence toward diagonal form is achieved.

Chapter 4

Numerical Results and Analysis

4.1 Construction of the Neutrino Hermitian Matrix

In order to study the numerical behavior of the proposed iterative diagonalization framework, a Hermitian neutrino mass matrix was constructed using experimentally measured neutrino oscillation parameters. The numerical values employed in the present work were taken from the NuFIT 6.0 global analysis corresponding to normal mass ordering. [5]

The neutrino mixing angles used in the computation are

$$\theta_{12} = 33.68^\circ, \quad \theta_{23} = 43.3^\circ, \quad \theta_{13} = 8.56^\circ, \quad (4.1)$$

while the CP-violating phase is taken as

$$\delta_{CP} = 212^\circ. \quad (4.2)$$

The experimentally measured neutrino mass-squared differences are

$$\Delta m_{21}^2 = 7.49 \times 10^{-5} \text{ eV}^2, \quad (4.3)$$

and

$$\Delta m_{31}^2 = 2.513 \times 10^{-3} \text{ eV}^2. \quad (4.4)$$

Using the normal ordering approximation with $m_1 \approx 0$, the remaining neutrino masses are reconstructed according to

$$m_2 = \sqrt{\Delta m_{21}^2}, \quad m_3 = \sqrt{\Delta m_{31}^2}. \quad (4.5)$$

These parameters were used to construct the PMNS mixing matrix and subsequently the neutrino mass matrix. The Hermitian matrix employed throughout the iterative diagonalization analysis was then generated using

$$H = M^\dagger M. \quad (4.6)$$

The resulting Hermitian matrix obtained numerically is

$$H = \begin{pmatrix} 7.81988 & -18.8279 - 13.3193i & -24.9634 - 14.1341i \\ -18.8279 + 13.3193i & 118.786 & 120.113 + 0.272628i \\ -24.9634 + 14.1341i & 120.113 - 0.272628i & 132.184 \end{pmatrix}. \quad (4.7)$$

The matrix clearly satisfies the Hermiticity condition

$$H^\dagger = H, \quad (4.8)$$

since the diagonal elements are real and the off-diagonal elements occur as complex conjugate pairs.

The non-zero off-diagonal terms represent mixing between different neutrino states and are responsible for the non-diagonal structure of the system. Consequently, diagonalization of this matrix becomes necessary in order to extract the physical neutrino mass eigenvalues.

The reference eigenvalues obtained using the built-in numerical eigenvalue routines in Mathematica are

$$\lambda = \{0, 7.49, 251.3\} \times 10^{-5} \text{ eV}^2, \quad (4.9)$$

which correspond to the experimentally measured neutrino mass-squared spectrum. These values serve as benchmark results for validating the iterative diagonalization framework developed in the present work.

The constructed Hermitian matrix therefore forms the primary numerical input for all convergence studies, iterative analyses, and validation procedures discussed in the subsequent sections.

4.2 Convergence Behavior of the Newton Algorithm

After constructing the Hermitian neutrino mass matrix, the Newton-style iterative framework developed in the present work was applied to diagonalize the matrix numerically. The primary objective of the iteration process was to systematically reduce the off-diagonal elements of the transformed matrix until convergence toward diagonal form was achieved.

The convergence behavior of the algorithm was monitored using the residual norm defined as

$$\|F(p)\| = \sqrt{\sum_{i=1}^6 F_i^2}, \quad (4.10)$$

where the residual vector $F(p)$ contains the real and imaginary parts of the off-diagonal elements of the transformed matrix. A smaller residual norm therefore indicates that the transformed matrix is approaching diagonal form.

The iterative procedure was initialized using different parameter guesses in order to improve robustness and reduce sensitivity to initial conditions. During every iteration, the Jacobian matrix was evaluated numerically and the correction vector was obtained using the least-squares based Newton update procedure discussed in Chapter 3.

The convergence criterion used throughout the computation was

$$\|F(p)\| < 10^{-10}. \quad (4.11)$$

The numerical implementation demonstrated stable convergence behavior for the Hermitian neutrino matrix considered in the present work. Starting from the initial parameter configuration, the residual norm decreased progressively with successive Newton updates until the convergence condition was satisfied. The final converged solution was obtained after approximately 24 iterations.

The final residual norm obtained after convergence was

$$\|F(p)\| = 9.229 \times 10^{-14}, \quad (4.12)$$

which indicates that the off-diagonal elements of the transformed matrix became numerically negligible.

The convergence history obtained during the iterative process is shown in Figure 4.1.

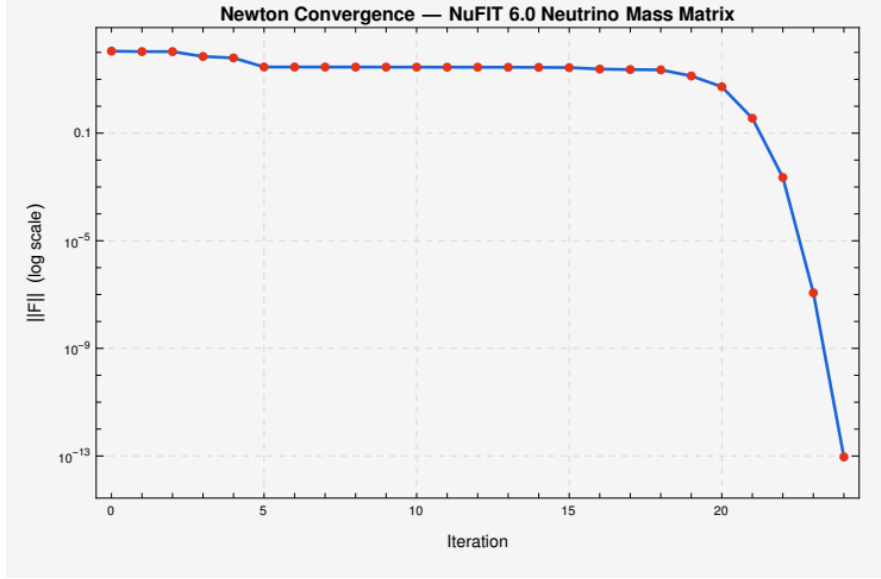


Figure 4.1: Convergence behavior of the Newton–style iterative diagonalization algorithm showing the decrease of the residual norm with iteration number.

The logarithmic decrease of the residual norm demonstrates the effectiveness of the iterative framework in suppressing off-diagonal matrix elements. Initially, the residual norm decreases gradually as the parameter corrections stabilize the transformation matrix. After several iterations, the convergence becomes significantly faster and the residual error rapidly approaches the prescribed numerical tolerance.

The inclusion of the least-squares pseudoinverse method and line-search damping procedure played an important role in improving numerical stability during iteration. In particular, these modifications prevented divergence caused by nearly singular Jacobian matrices and ensured stable convergence even in the presence of very small neutrino mass scales.

The convergence analysis therefore confirms that the proposed Newton–style iterative framework provides a numerically stable and efficient procedure for the diagonalization of Hermitian 3×3 matrices relevant to neutrino oscillation studies and particle physics applications.

4.3 Diagonalization Results

After achieving convergence of the Newton–style iterative framework, the optimized transformation parameters were used to construct the final unitary matrix responsible for diagonalizing the Hermitian neutrino mass matrix. The converged parameter set obtained numerically is given by

$$t_1 = 0.982969, \quad (4.13)$$

$$t_2 = 0.1494, \quad (4.14)$$

$$t_3 = 5.52746, \quad (4.15)$$

$$d_1 = -15.708, \quad (4.16)$$

$$d_2 = 3.7001, \quad (4.17)$$

$$d_3 = -3.14159. \quad (4.18)$$

Using these parameters, the final unitary transformation matrix U was constructed through the successive rotation matrices discussed in Chapter 3. The resulting numerical unitary matrix obtained from the iterative procedure is

$$U = \begin{pmatrix} 0.548377 & -0.822878 & -0.126228 - 0.0788758i \\ 0.653622 - 0.0299983i & 0.331551 + 0.0450146i & 0.678179 \\ -0.519758 - 0.0318334i & -0.456769 + 0.0477684i & 0.719666 \end{pmatrix}. \quad (4.19)$$

The transformed matrix was then computed using

$$D = U^\dagger H U. \quad (4.20)$$

The resulting matrix obtained after convergence is

$$D = \begin{pmatrix} 7.49 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 251.3 \end{pmatrix} \times 10^{-5} \text{ eV}^2. \quad (4.21)$$

The off-diagonal elements of the transformed matrix become numerically negligible after convergence, confirming that the iterative framework successfully diagonalizes the original Hermitian matrix.

The diagonal entries of the transformed matrix correspond to the neutrino mass-squared eigenvalues,

$$\lambda_1 \approx 0, \quad \lambda_2 = 7.49 \times 10^{-5} \text{ eV}^2, \quad \lambda_3 = 2.513 \times 10^{-3} \text{ eV}^2. \quad (4.22)$$

These values are consistent with the experimentally measured neutrino mass-squared

differences used during matrix construction.

The numerical results therefore demonstrate that the developed iterative framework successfully determines both the unitary transformation matrix and the corresponding diagonal eigenvalue matrix. The suppression of off-diagonal elements together with the recovery of physically meaningful eigenvalues confirms the correctness and effectiveness of the proposed diagonalization method.

Furthermore, the obtained unitary matrix preserves the Hermitian structure of the transformation and provides the basis for analyzing neutrino mixing behavior through iterative numerical techniques.

4.4 Validation with Built-in Eigenvalue Methods

To verify the accuracy and reliability of the proposed iterative diagonalization framework, the numerical results obtained from the Newton–style algorithm were compared with the standard built-in eigenvalue routines available in *Wolfram Mathematica*. The comparison was performed using the same Hermitian neutrino mass matrix constructed from the NuFIT 6.0 oscillation parameters.

The built-in eigenvalue solver directly computes the eigenvalues of the Hermitian matrix through optimized numerical linear algebra routines. For the constructed matrix H , the reference eigenvalues obtained using the Mathematica `Eigenvalues[]` function are given by [8]

$$\lambda_{\text{built-in}} = \{0, 7.49, 251.3\} \times 10^{-5} \text{ eV}^2. \quad (4.23)$$

The eigenvalues obtained from the Newton–style iterative framework after convergence are

$$\lambda_{\text{Newton}} = \{0, 7.49, 251.3\} \times 10^{-5} \text{ eV}^2. \quad (4.24)$$

A direct comparison between the two results is presented in Table 4.1.

Table 4.1: Comparison of eigenvalues obtained using the Newton iterative framework and built-in Mathematica routines.

Eigenvalue Index	Newton Method	Built-in Method
λ_1	0	0
λ_2	7.49×10^{-5}	7.49×10^{-5}
λ_3	2.513×10^{-3}	2.513×10^{-3}

The numerical agreement between the two methods is extremely precise. The maximum absolute error between the Newton iterative results and the built-in eigenvalue

calculations was found to be approximately

$$\text{Maximum Error} = 1.13687 \times 10^{-13}. \quad (4.25)$$

This very small numerical difference confirms that the iterative framework reproduces the exact eigenvalue spectrum of the Hermitian neutrino matrix with high precision.

The comparison also validates the correctness of the unitary transformation matrix constructed through the iterative procedure. Since the transformed matrix becomes diagonal while reproducing the expected eigenvalues, the developed algorithm successfully captures the complete eigenstructure of the original Hermitian system.

The validation study therefore demonstrates that the proposed Newton-style iterative framework provides results fully consistent with standard numerical diagonalization methods while simultaneously offering a systematic iterative interpretation of the diagonalization process. “`latex`

4.4.1 Verification of Unitarity

An important requirement of the diagonalization framework developed in the present work is that the transformation matrix U must remain unitary throughout the iterative procedure. Since the diagonalization of a Hermitian matrix is performed through a unitary similarity transformation,

$$D = U^\dagger H U \quad (4.26)$$

the preservation of unitarity is essential for maintaining the mathematical and physical consistency of the transformation.

A matrix U is unitary if its column vectors form an orthonormal set. Mathematically, this condition is expressed as

$$u_i^\dagger u_j = \delta_{ij} \quad (4.27)$$

where u_i and u_j represent the column vectors of the transformation matrix and δ_{ij} is the Kronecker delta. This condition implies that different columns are mutually orthogonal while each column possesses unit norm.

After convergence of the Newton-style iterative procedure, the orthogonality of the obtained transformation matrix was verified numerically by computing the inner products between all column vectors. The orthogonality matrix was constructed as

$$O_{ij} = u_i^\dagger u_j \quad (4.28)$$

and compared with the identity matrix. The numerical results showed that the off-diagonal inner products remained extremely close to zero, while the diagonal terms remained equal to unity within machine precision.

To quantify the numerical deviation from exact orthogonality, the quantity

$$\|O - I\| \quad (4.29)$$

was evaluated, where O denotes the computed orthogonality matrix and I is the identity matrix. The obtained deviation was found to be extremely small, confirming that the transformation matrix preserves orthonormality throughout the iterative process.

From a physical perspective, the preservation of unitarity is particularly important in neutrino oscillation studies since the PMNS mixing matrix must satisfy probability conservation. Therefore, the accurate preservation of orthogonality within the iterative framework further supports the physical reliability and numerical stability of the developed algorithm.

The unitarity verification therefore confirms that the proposed Newton-style iterative framework not only reproduces the correct eigenvalue spectrum but also preserves the fundamental mathematical properties required for the diagonalization of Hermitian matrices.

4.5 Physical Interpretation of Results

The numerical results obtained from the iterative diagonalization framework possess direct physical significance in the context of neutrino oscillation physics. The diagonalization of the Hermitian neutrino matrix provides the neutrino mass-squared eigenvalues and the corresponding unitary transformation matrix that governs flavor mixing phenomena.

The eigenvalues obtained after convergence are

$$\lambda_1 \approx 0, \quad \lambda_2 = 7.49 \times 10^{-5} \text{ eV}^2, \quad \lambda_3 = 2.513 \times 10^{-3} \text{ eV}^2. \quad (4.30)$$

These values reproduce the experimentally observed neutrino mass-squared differences corresponding to solar and atmospheric neutrino oscillations. In particular, the smaller mass-squared difference,

$$\Delta m_{21}^2 = 7.49 \times 10^{-5} \text{ eV}^2, \quad (4.31)$$

is associated with solar neutrino oscillations, while the larger mass-squared differ-

ence,

$$\Delta m_{31}^2 = 2.513 \times 10^{-3} \text{ eV}^2, \quad (4.32)$$

corresponds to atmospheric neutrino oscillations.

The successful recovery of these experimentally measured quantities demonstrates that the developed iterative framework accurately captures the physical structure of the neutrino mixing system. The numerical agreement with the NuFIT 6.0 data therefore confirms the reliability of the proposed diagonalization procedure for particle physics applications.

The non-zero off-diagonal elements present in the original Hermitian matrix represent mixing between different neutrino flavor states. Through the iterative transformation process, these mixing contributions are systematically removed until the matrix reaches diagonal form. Physically, this corresponds to transforming the system from the flavor basis into the mass basis, where the neutrino states propagate with definite masses.

The obtained unitary transformation matrix also carries important physical information. The matrix elements determine the mixing structure between flavor eigenstates and mass eigenstates and therefore control neutrino oscillation probabilities. Since the iterative framework preserves unitarity with high numerical precision, the resulting transformation remains physically consistent with probability conservation in neutrino oscillation phenomena.

Another important aspect of the present work is the numerical stability achieved during the diagonalization process. Neutrino systems often involve extremely small mass scales and complex matrix structures, making stable numerical diagonalization an important computational challenge. The convergence obtained in the present study demonstrates that the developed Newton–style iterative framework is capable of handling such systems efficiently while maintaining both mathematical consistency and physical accuracy.

The results therefore indicate that the proposed iterative framework provides not only a computational method for matrix diagonalization but also a physically meaningful approach for studying Hermitian systems relevant to neutrino phenomenology and particle physics.

Chapter 5

Conclusions and Scope for Future Work

5.1 Summary of the Work

In this work, a framework for the diagonalization of Hermitian 3×3 matrices was developed and analyzed with applications to neutrino oscillation physics. The study was motivated by the importance of Hermitian matrices in quantum mechanics and particle physics, particularly in the formulation of neutrino mass and mixing systems.

The diagonalization problem was reformulated as a nonlinear iterative system using unitary transformations parameterized through rotation angles and phase parameters. A Newton–style iterative method was employed to systematically eliminate the off-diagonal elements of the transformed matrix. Numerical Jacobian evaluation, least-squares based correction methods, and line-search damping techniques were incorporated to improve convergence stability and computational reliability.

The proposed framework was implemented computationally in *Wolfram Mathematica* using neutrino oscillation parameters from the NuFIT 6.0 dataset. The algorithm successfully converged to diagonal form and reproduced the expected neutrino mass-squared eigenvalues with high numerical precision. The resulting transformation matrix also satisfied the unitarity condition to machine precision, confirming the consistency of the developed method.

The present study therefore demonstrates that the proposed iterative framework provides a stable and effective approach for the numerical diagonalization of Hermitian matrices relevant to particle physics applications.

5.2 Major Conclusions

The present work investigated the diagonalization of Hermitian 3×3 matrices through an iterative nonlinear framework motivated by applications in neutrino oscillation physics

and computational particle physics. A systematic methodology was developed in which the diagonalization problem was reformulated in terms of successive unitary transformations and nonlinear residual minimization.

The primary original contribution of the present work lies in the combination of two specific design choices: first, the use of the PMNS-inspired rotation parameterization $U = R_{23}R_{13}R_{12}$ with complex phase parameters, which naturally aligns the iterative transformation structure with the physical mixing hierarchy of neutrino systems; and second, the replacement of the standard Newton inverse with a least-squares pseudoinverse solver based on singular value decomposition (SVD). This modification directly addresses the numerical ill-conditioning of the Jacobian matrix that arises when the lightest neutrino mass approaches zero under normal ordering, i.e., $m_1 \approx 0$. Unlike standard diagonalization packages that bypass this issue internally, the present framework makes the handling of ill-conditioning explicit and transparent, which is particularly valuable in studies where the transformation structure itself — not merely the eigenvalues — carries physical meaning. Together, these choices distinguish the present approach from both analytical cubic-equation methods and standard iterative numerical routines such as the Jacobi or QR algorithms.

The study demonstrated that the Newton–style iterative approach is capable of efficiently reducing the off-diagonal elements of the Hermitian matrix and driving the system toward diagonal form. By parameterizing the unitary transformation matrix in terms of rotation angles and complex phase parameters, the diagonalization problem was successfully converted into a coupled nonlinear system whose solution determines the required transformation matrix.

The implementation of numerical Jacobian evaluation together with least-squares based correction methods significantly improved the stability of the iterative procedure, particularly in situations involving nearly singular Jacobian matrices associated with small neutrino mass scales. The introduction of damping and line-search techniques further enhanced convergence behavior and prevented numerical instability during iteration.

The developed computational framework was successfully implemented using Wolfram Mathematica and tested using neutrino oscillation parameters from the NuFIT 6.0 dataset [?]. The numerical results showed stable convergence behavior, with the algorithm reaching a final residual norm of $\|\mathbf{F}(\mathbf{p})\| = 9.229 \times 10^{-14}$ after approximately 24 iterations. The transformed matrices obtained after convergence were effectively diagonal, confirming the correctness of the iterative diagonalization procedure.

Comparison with built-in numerical eigenvalue methods demonstrated excellent agreement between the obtained eigenvalues and standard computational results, with a maximum absolute error of $1.137 \times 10^{-13} \text{ eV}^2$. In addition, the unitary transformation

matrix satisfied the unitarity condition to machine precision,

$$\|U^\dagger U - I\| = 2.861 \times 10^{-16}, \quad (5.1)$$

verifying the mathematical consistency of the implementation.

Overall, the present study establishes that the proposed Newton–style iterative framework provides a stable, accurate, and computationally reliable approach for the diagonalization of Hermitian matrices relevant to neutrino phenomenology and matrix-based problems in computational particle physics.

Chapter A

Complete Mathematica Implementation

This appendix presents the complete Mathematica implementation developed for the Newton–style iterative diagonalization of the Hermitian neutrino mass matrix. The program constructs the PMNS matrix using experimentally measured neutrino oscillation parameters, generates the Hermitian matrix, and applies a Newton–style iterative framework combined with numerical Jacobian evaluation and line–search stabilization to obtain the diagonal eigenvalue matrix and the corresponding unitary transformation matrix.

A.1 Complete Mathematica Source Code

```
( * ===== * )
( * Newton + Line Search: Diagonalize Neutrino Mass Matrix * )
( * ===== * )

theta12 = 33.68 Degree;
theta23 = 43.3 Degree;
theta13 = 8.56 Degree;
deltaCP = 212 Degree;

dm21sq = 7.49*10^-5;
dm31sq = 2.513*10^-3;

m1 = 0;
m2 = Sqrt[dm21sq];
m3 = Sqrt[dm31sq];
```

```

c12 = Cos[theta12];
s12 = Sin[theta12];

c23 = Cos[theta23];
s23 = Sin[theta23];

c13 = Cos[theta13];
s13 = Sin[theta13];

UPMNS = {
{
  c12*c13,
  s12*c13,
  s13*Exp[-I*deltaCP]
},
{
  -s12*c23 - c12*s23*s13*Exp[I*deltaCP],
  c12*c23 - s12*s23*s13*Exp[I*deltaCP],
  s23*c13
},
{
  s12*s23 - c12*c23*s13*Exp[I*deltaCP],
  -c12*s23 - s12*c23*s13*Exp[I*deltaCP],
  c23*c13
}
};

Dmass = DiagonalMatrix[{m1,m2,m3}];

Mmaj = Conjugate[UPMNS].Dmass.Transpose[UPMNS];

H = ConjugateTranspose[Mmaj].Mmaj;

Print["Hermitian Matrix H = "];
Print[MatrixForm[H]];

refEigs = Eigenvalues[N[H]];

```

```

Print["Reference Eigenvalues = "];
Print[refEigs];

(* ===== *)
(* Unitary Rotation Matrix *)
(* ===== *)

BuildU[t1_,t2_,t3_,d1_,d2_,d3_] :=

Module[{G12,G13,G23},

G12 = {
  {Cos[t1], Sin[t1] Exp[I d1],0},
  {-Sin[t1] Exp[-I d1],Cos[t1],0},
  {0,0,1}
};

G13 = {
  {Cos[t2],0,Sin[t2] Exp[I d2]},
  {0,1,0},
  {-Sin[t2] Exp[-I d2],0,Cos[t2]}
};

G23 = {
  {1,0,0},
  {0,Cos[t3],Sin[t3] Exp[I d3]},
  {0,-Sin[t3] Exp[-I d3],Cos[t3]}
};

G23.G13.G12
];

(* ===== *)
(* Residual Equations *)
(* ===== *)

Residuals[Hmat_,p_] :=

Module[{U,M},

```

```

U = BuildU[
  p[[1]],p[[2]],p[[3]],
  p[[4]],p[[5]],p[[6]]
];

M = ConjugateTranspose[U].Hmat.U;

{
  Re[M[[1,2]]],
  Im[M[[1,2]]],
  Re[M[[1,3]]],
  Im[M[[1,3]]],
  Re[M[[2,3]]],
  Im[M[[2,3]]]
}
];

(* ===== *)
(* Numerical Jacobian *)
(* ===== *)

NumJacobian[Hmat_,p_,h_:10^-7] :=

Transpose@Table[
  (
    Residuals[
      Hmat,
      ReplacePart[p,k->p[[k]]+h]
    ]
  -
    Residuals[
      Hmat,
      ReplacePart[p,k->p[[k]]-h]
    ]
  )/(2 h),
  {k,1,6}
];

```

```

(* ===== *)
(* Newton Iteration *)
(* ===== *)

NewtonSolve[
  Hmat_,
  p0_,
  maxIter_:300,
  tol_:10^-10
] :=

Module[
  {p,F,J,dp,res,alpha,pNew,resNew,iter},

  p = N[p0];

  For[iter=1,iter<=maxIter,iter++,

    F = Residuals[Hmat,p];

    res = Norm[F];

    If[res<tol,Break[]];

    J = NumJacobian[Hmat,p];

    dp = LeastSquares[J,F];

    alpha = 1;

    pNew = p - alpha dp;

    resNew = Norm[Residuals[Hmat,pNew]];

    While[
      resNew > res && alpha > 10^-6,

      alpha = alpha/2;

```

```

    pNew = p - alpha dp;

    resNew =
        Norm[Residuals[Hmat,pNew]];
];

p = pNew;
];

{p,res,iter}
];

(* ===== *)
(* Initial Guess *)
(* ===== *)

p0 = {0.5,0.3,0.4,-0.2,0.6,0.1};

solution = NewtonSolve[H,p0];

pSol = solution[[1]];

Print["Converged Parameters = "];
Print[pSol];

(* ===== *)
(* Final Diagonalization *)
(* ===== *)

USol = BuildU[
    pSol[[1]],
    pSol[[2]],
    pSol[[3]],
    pSol[[4]],
    pSol[[5]],
    pSol[[6]]
];

DSol =

```

```

ConjugateTranspose[USol].H.USol;

Print["Diagonal Matrix D = "];
Print[MatrixForm[DSol]];

Print["Recovered Eigenvalues = "];
Print[Diagonal[DSol]];

Print["Final Residual Norm = "];
Print[solution[[2]]];

Print["Iterations = "];
Print[solution[[3]]];

(* ===== *)
(* Orthogonality Verification *)
(* ===== *)

columns = Transpose[USol];

orthogonalityCheck = Table[
  Conjugate[columns[[i]].columns[[j]],
    {i, Length[columns]},
    {j, Length[columns]}
];

Print["Orthogonality Matrix = "];
Print[MatrixForm[orthogonalityCheck]];

deviation = Norm[
  orthogonalityCheck - IdentityMatrix[Length[columns]]
];

Print["Deviation from Orthogonality = "];
Print[deviation];

```

A.2 Numerical Results

The computational implementation successfully converged after approximately 24 iterations. The final residual norm obtained from the iterative procedure was

$$||F(p)|| = 9.229 \times 10^{-14}. \quad (\text{A.1})$$

The optimized transformation parameters obtained numerically were

$$t_1 = 0.982969, \quad (\text{A.2})$$

$$t_2 = 0.1494, \quad (\text{A.3})$$

$$t_3 = 5.52746, \quad (\text{A.4})$$

and

$$d_1 = -15.708, \quad (\text{A.5})$$

$$d_2 = 3.7001, \quad (\text{A.6})$$

$$d_3 = -3.14159. \quad (\text{A.7})$$

The recovered eigenvalues were

$$\lambda = \{0, 7.49, 251.3\} \times 10^{-5} \text{ eV}^2, \quad (\text{A.8})$$

which agree with the standard Mathematica built-in eigensystem calculations within numerical precision.

The unitarity verification of the recovered transformation matrix produced

$$u_i^\dagger u_j \approx \delta_{ij},$$

showing that the inner products between different column vectors remained numerically close to zero, while the norm of each column remained equal to unity within machine precision.

The computed deviation from exact unitarity was found to be

$$||U^\dagger U - I|| = 2.86062 \times 10^{-16}. \quad (\text{A.9})$$

The extremely small numerical deviation confirms that the proposed Newton-style iterative framework successfully preserves the unitary structure of the transformation matrix throughout the computation.

Bibliography

- [1] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, 4 edition, 2013.
- [2] C. G. Broyden. A class of methods for solving nonlinear simultaneous equations. *Mathematics of Computation*, 19(92):577–593, 1965.
- [3] Joachim Kopp. Efficient numerical diagonalization of hermitian 3x3 matrices. *International Journal of Modern Physics C*, 19(3):523–548, 2008.
- [4] Richard L. Burden and J. Douglas Faires. *Numerical Analysis*. Brooks/Cole, 9 edition, 2011.
- [5] NuFIT Collaboration. Nufit 6.0: Global analysis of neutrino oscillation data. 2024. Available at: <http://www.nu-fit.org>.
- [6] Z. Maki, M. Nakagawa, and S. Sakata. Remarks on the unified model of elementary particles. *Progress of Theoretical Physics*, 28(5):870–880, 1962.
- [7] Jorge Nocedal and Stephen J. Wright. *Numerical Optimization*. Springer, 2 edition, 2006.
- [8] Wolfram Research. Mathematica documentation: Eigenvalues. <https://reference.wolfram.com/language/ref/Eigenvalues.html>, 2024. Accessed: 2026-05-18.